	Application No.	**Applicant(s)**
Examiner-Initiated Interview Summary	10/696,968	LAURA, JOSEPH G.
	Examiner	**Art Unit**	
	Eric B. Kiss	2192	

All Participants:

(1) *Eric B. Kiss*.

(2) *Brian Genco (Reg. No. 58,095)*.

Status of Application: _____

(3) *Michael W. Piper (Reg. No. 39,800)**.

(4) _____.

Date of Interview: *12 September 2007*   **Time:** _____

Type of Interview:
☒ Telephonic
☐ Video Conference
☐ Personal (Copy given to: ☐ Applicant   ☐ Applicant's representative)

Exhibit Shown or Demonstrated: ☐ Yes   ☒ No
If Yes, provide a brief description: .

Part I.

Rejection(s) discussed:

Claims discussed:
1,8,12,13,16,17,18,23,29,30,31,33,35,37,40

Prior art documents discussed:

Part II.

SUBSTANCE OF INTERVIEW DESCRIBING THE GENERAL NATURE OF WHAT WAS DISCUSSED:
See Continuation Sheet

Part III.

☒ It is not necessary for applicant to provide a separate record of the substance of the interview, since the interview directly resulted in the allowance of the application. The examiner will provide a written summary of the substance of the interview in the Notice of Allowability.

☐ It is not necessary for applicant to provide a separate record of the substance of the interview, since the interview did not result in resolution of all issues. A brief summary by the examiner appears in Part II above.

_____   _____
(Examiner/SPE Signature)   (Applicant/Applicant's Representative Signature – if appropriate)

Continuation of Substance of Interview including description of the general nature of what was discussed: The examiner proposed amendments to the claims to place the application in clear condition for allowance. In particular, the examiner proposed amendments to more clearly set forth in the claims the implementation of queues and shared memory in the COBOL programs. Mr. Genco drafted a copy of the amended claims consistent with the discussion. *Although Mr. Piper did not personally participate in the telephone conversations leading to the drafted amendment, he personally reviewed and signed the amendment, and authorized its entry to place the application in condition for allowance. Attached to this Interview Summary are pages 1 and 13 of the amendment faxed to the examiner for entry. Pages 2 through 12 are incorporated into the Examiner's Amendment itself.

ESK

SUMMARY 09/12/2007

INTERVIEW

ATTACHMENT TO

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Joseph G. Laura	§
		§ Group Art Unit: 2192
Serial No.:	10/696,968	§
		§ Examiner: Kiss, Eric B.
Filed:	October 30, 2003	§
		§ Confirmation No.: 8504
For:	SYSTEM AND METHOD FOR	§
	COBOL TO PROVIDE SHARED	§
	MEMORY AND MEMORY AND	§
	MESSAGE QUEUES	§

Applicant thanks Primary Examiner Eric Kiss for the telephone call on September 11, 2007. In the telephone call, Primary Examiner Eric Kiss discussed proposed claim amendments that would appear to place the application in a condition for allowance. The following amendments follow the suggestions provided by Primary Examiner Eric Kiss. The changes made are shown by underlining the added text and striking through the deleted text.

1

Listing of the Claims:

1. (Currently Amended)   A system to enable queues for COBOL programs, the system comprising:

> a memory with a memory space operable for a queue;

> an operating system stored on a computer readable medium maintaining a key and an address of the memory space related to the key;

> a COBOL routine stored on a computer readable medium maintaining the key in an index, the COBOL routine communicating with the operating system to receive the address of the memory space based on the key;

> a COBOL program stored on a computer readable medium communicating with the COBOL routine to receive the memory address based on the key in the index maintained by the COBOL routine;

> a coordination module stored on a computer readable medium coordinating reading and writing information to the queue.

2. (Previously Presented) The system of Claim 1, wherein the COBOL program communicates with the COBOL routine through a call to the COBOL routine using an identifier and wherein the index maintains the identifier associated with the key.

3. (Original) The system of Claim 2, wherein the index maintains a plurality of identifiers each associated with one of a plurality of keys maintained by the index.

4. (Original) The system of Claim 3, wherein the plurality of identifiers are further

defined as an alphanumeric identifier.

5. (Original) The system of Claim 3, wherein the plurality of identifiers are further

defined as names.

6. (Original) The system of Claim 1, wherein the COBOL program receives the

memory address via a linkage section of the COBOL program.

7. (Original) The system of Claim 1, wherein the COBOL program is operable to

receive the memory address to enable the COBOL program to resolve the

information in the memory space to the linkage section of the COBOL program.

8. (Currently Amended)   The system of claim 1, wherein the ~~queue~~ queue ~~memory space~~ is

~~operable for~~ a message queue.

9. (Currently Amended)   The system of Claim 8, ~~further comprising a~~ wherein the

coordination module is operable to receive a request from the COBOL program to

read and write information to the message queue.

10. (Original) The system of Claim 9, wherein the coordination module coordinates

reading and writing information to the message queue in a last-in-first-out order.

11. (Original) The system of Claim 9, wherein the coordination module coordinates

3

reading and writing information to the message queue in a first-in-first-out order.

12. (Currently Amended)   The system of Claim 1, wherein the queue ~~memory space~~ is ~~operable for~~ a memory queue.

13. (Currently Amended)   The system of Claim 12, ~~further comprising a~~ wherein the coordination module is operable to prevent a conflict.

14. (Original) The system of Claim 13, wherein the coordination module is operable from a call to an operating system.

15. (Original) The system of Claim 13, wherein the coordination module is operable to prevent writing when the memory space is full and further operable to prevent reading when the memory space is empty.

4

46396.03/4000.15900

PAGE 4/13 * RCVD AT 9/12/2007 8:37:34 AM [Eastern Daylight Time] * SVR:USPTO-EFXRF-2/10 * DNIS:2733699 * CSID:972 731 2289 * DURATION (mm-ss):03-00

16. (Currently Amended) A method of enabling queues for COBOL programs,

comprising:

creating a queue using a memory space;

maintaining a key and an address of the memory space related to the key with

an operating system stored on a computer readable medium;

maintaining the key in an index in association with an identifier;

communicating with the operating system to receive the address of the memory

space based on the key from the index;

resolving the memory space to an operable portion of a COBOL program

stored on a computer readable medium based on the address of the

memory space retrieved in accordance with the identifier associated with

the key from the index; and

wherein at least one COBOL routine stored on a computer readable medium

maintains the index and receives the address of the memory space from

the operating system and coordinates reading and writing information to

the queue.

17. (Canceled)

18. (Currently Amended)   The method of Claim 16[[17]], wherein the COBOL routine is

further defined as a COBOL technical layer having a plurality of routines, the

5

method further comprising:

attaching to an existing queue;

querying the queue to determine whether the queue exists and to determine the size of the queue;

adding, by a push module of the COBOL technical layer, at least one row to the queue;

blocking when the queue is full;

removing, by a pop module of the COBOL technical layer, a top row from the queue;

detaching from a queue; and

removing a queue from a system.

19. (Original) The method of Claim 18, wherein the COBOL technical layer is further defined as a COBOL library wherein the routines are callable from the COBOL program.

20. (Original) The method of Claim 18, wherein the COBOL technical layer is integral to the COBOL program.

21. (Original) The method of Claim 18, wherein the COBOL technical layer is further defined as enabled by a COBOL compiler.

22. (Original) The method of Claim 21, wherein the compiler enabled functionality is

6

further defined as pre-compiler enabled functionality.

7

23. (Currently Amended)   A method of sharing memory between COBOL programs, comprising:

communicating a call to an operating system stored on a computer readable medium for a block of memory;

communicating a first request by a first COBOL program stored on a computer readable medium for an address of the block of memory;

communicating a second request by a second COBOL program stored on a computer readable medium for an address of the block of memory;

returning the address to a linkage section of the first COBOL program in response to the first request;

returning the address to a linkage section of the second COBOL program in response to the second request.; and

sharing, by the first and second COBOL programs, the block of memory.

wherein the first and second COBOL programs use the address to map the linkage sections of each of the first and second COBOL programs to the block of memory to enable the first and second COBOL programs to share the block of memory.

24. (Previously Presented) The method of Claim 23, wherein the operating system allocates the block of memory in response to the call.

8

46396.03/4000.15900

PAGE 8/13 * RCVD AT 9/12/2007 8:37:34 AM [Eastern Daylight Time] * SVR:USPTO-EFXRF-2/10 * DNIS:2733699 * CSID:972 731 2289 * DURATION (mm-ss):03-00

Attorney Docket No: IDF 2562 (4000-15900)   *Patent*

25. (Previously Presented) The method of Claim 24, wherein the operating system allocates the block of memory based on parameters obtained from one of the first and second COBOL programs at compilation.

26. (Previously Presented) The method of Claim 24, wherein the operating system maintains the address of the block of memory and returns the address in response to the first and second requests.

27. (Previously Presented) The method of Claim 23, wherein the first request includes an identifier of the block of memory and the second request includes the identifier of the block of memory.

28. (Previously Presented) The method of Claim 27, further comprising maintaining an index including the identifier and a key associated with the identifier, the operating system using the key to return the address of the block of memory.

29. (Canceled)

9

30. (Currently Amended)   A method of sharing memory between COBOL programs, the method comprising:

maintaining, by a COBOL routine stored on a computer readable medium, an index of shared memory addresses;

requesting, by a <u>first</u> COBOL program stored on a computer readable medium, a shared memory block; ~~and~~

receiving to a linkage section of the <u>first</u> COBOL program an address of the shared memory block from the COBOL routine<u>;</u>

<u>using the address of the shared memory block to map the linkage section of the first COBOL program to the shared memory block;</u>

<u>requesting, by a second COBOL program stored on a computer readable medium, the shared memory block;</u>

<u>receiving to a linkage section of the second COBOL program the address of the shared memory block from the COBOL routine; and</u>

<u>using the address of the shared memory block to map the linkage section of the second COBOL program to the shared memory, thereby creating a shared block of memory useable by both the first and second COBOL programs.</u>

31. (Currently Amended)   The method of Claim 30, wherein the COBOL routine is a part of at least the <u>first or second</u> COBOL program.

10

Attorney Docket No: IDF 2562 (4000-15900)   *Patent*

32. (Original) The method of Claim 30, wherein the COBOL routine further comprises a plurality of COBOL subroutines.

33. (Currently Amended)   The method of Claim 30, wherein the COBOL routine is a library routine callable from the <u>first or second</u> COBOL program.

34. (Original) The method of Claim 30, wherein the COBOL routine is a COBOL function enabled by a COBOL compiler.

35. (Currently Amended)   The method of Claim 30, further comprising creating, by the COBOL routine, [[a]]<u>the</u> shared memory block.

36. (Original) The method of Claim 35, wherein creating the shared memory block further comprises calling the operating system to from the COBOL routine to request a block of memory.

37. (Currently Amended)   The method of Claim 36, <s>further comprising</s> <u>wherein mapping the linkage section of the first and second COBOL program to the shared memory further comprises</u> attaching the <u>first and second</u> COBOL program to the shared memory block.

38. (Original) The method of Claim 30, wherein the method further comprises maintaining, by the COBOL routine, an index having an identifier associated with the address of the shared memory block.

11

39. (Original) The method of Claim 38, wherein the method further comprises searching the index based on the identifier to locate the address of the shared memory block associated with the identifier.

40. (Currently Amended) The method of Claim 39, wherein the searching is accomplished by the COBOL routine in response to receiving a request from the first or the second COBOL programs, the request including the identifier.

41. (Currently Amended) The method of Claim 40, wherein the searching is accomplished by the first or the second COBOL programs.

42. (Original) The method of Claim 41, wherein the shared memory is defined as protected, and the method further comprises calling a semaphore routine to manage modifications to the shared memory.

43. (Previously Presented) The method of Claim 32, wherein the plurality of COBOL subroutines include:

creating the shared memory;

attaching to the shared memory;

detaching from the shared memory;

removing the shared memory; and

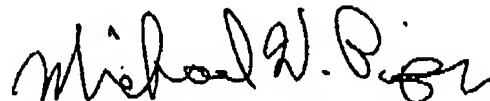querying the shared memory to determine whether the shared memory exists and a size of the shared memory.

12

Attorney Docket No: IDF 2562 (4000-15900) *Patent*

Conclusion

Applicant respectfully submits that the present application is in condition for allowance for the reasons stated above. If the Examiner has any questions or comments or otherwise feels it would be helpful in expediting the application, he is encouraged to telephone the undersigned at (972) 731-2288.

Respectfully submitted,

Date: ___September 11, 2007___

Michael W. Piper
Reg. No. 39,800

CONLEY ROSE, P.C.
5700 Granite Parkway, Suite 330
Plano, Texas 75024
(972) 731-2288
(972) 731-2289 (facsimile)

ATTORNEY FOR APPLICANT

13

46396.03/4000.15900